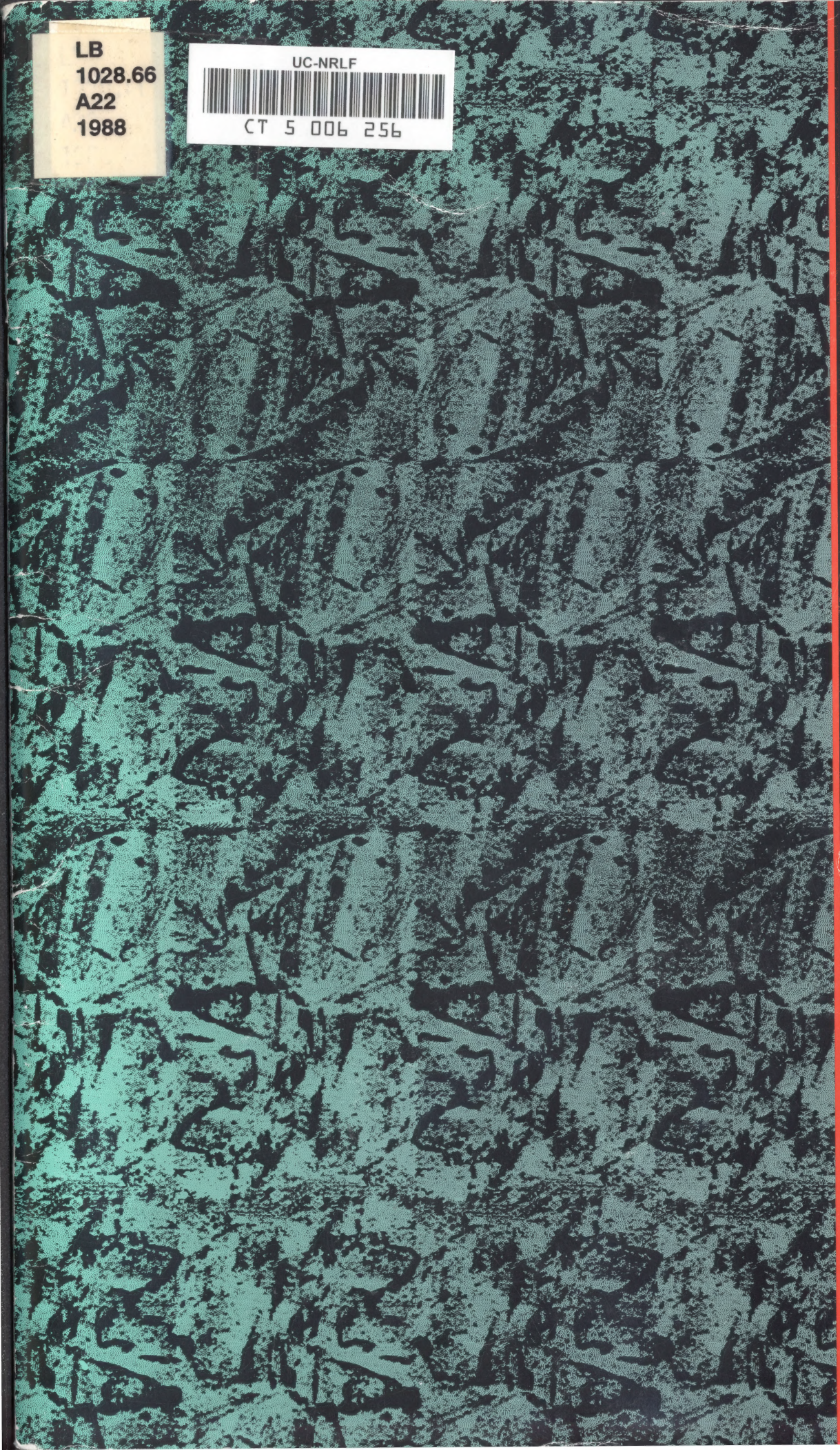


The
Academic
Courseware
Exchange

A Service
of Kinko's
Copies

*DEVELOPER'S
HANDBOOK*

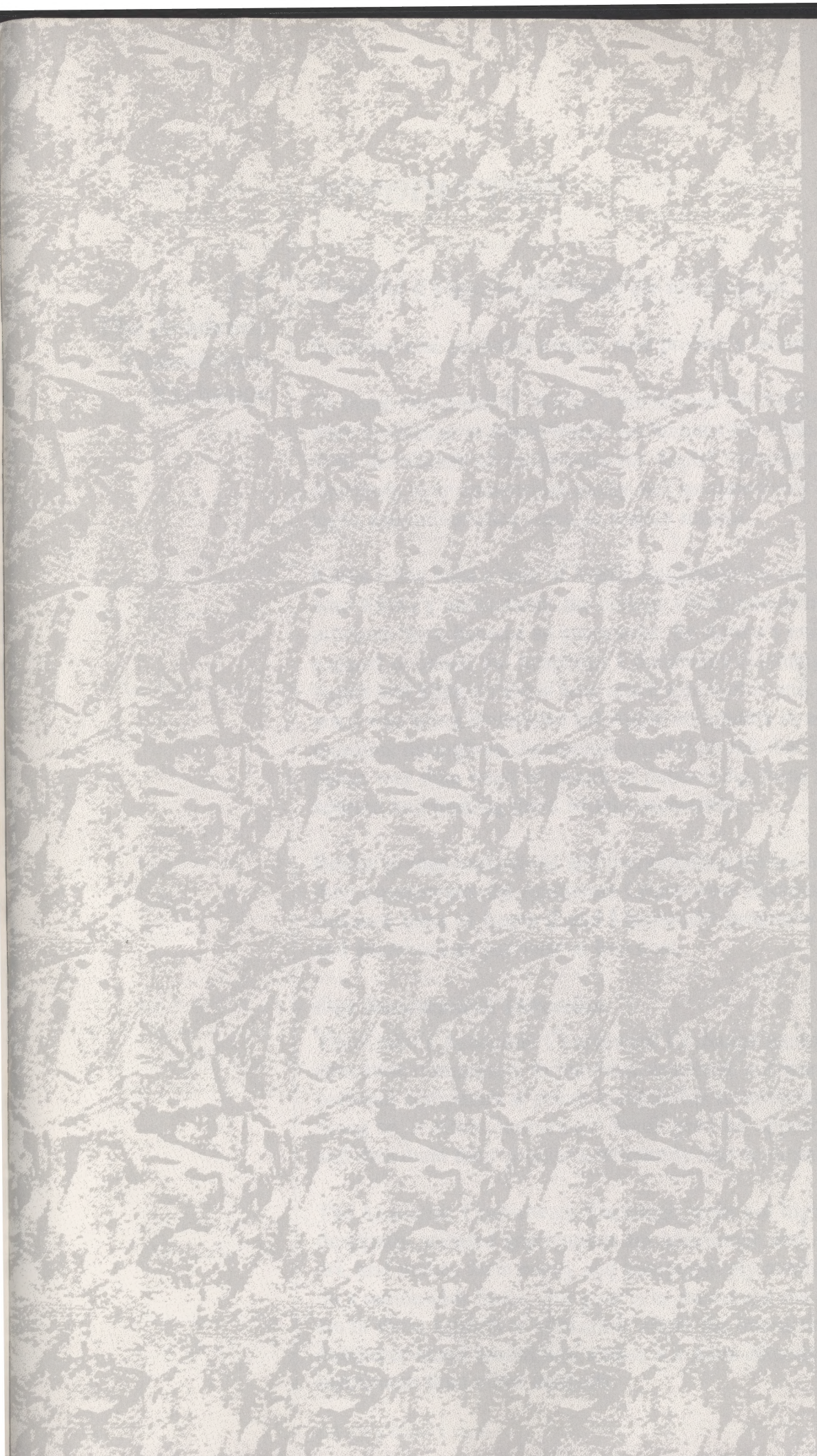
The
Academic
Courseware
Exchange

A Service
of Kinko's
Copies



*DEVELOPER'S*
*HANDBOOK*

# *TABLE OF CONTENTS*



# 1

# INTRODUCTION

# INTRODUCTION

## 1.1 The Academic Courseware Exchange

"I wonder if someone else could use my program."

Someone else probably could. And probably would like to.

To make that possible, Kinko's introduced the Academic Courseware Exchange at AppleWorld™ on January 15, 1986. For the first time, developers of courseware for Apple® computers had a way to distribute their courseware easily and inexpensively to other colleges and universities—and to realize returns.

Kinko's has been a user of Apple equipment for some time now. And Kinko's understands the college environment in which Apple products have become so successful. So Kinko's enlisted Apple's support to create the Academic Courseware Exchange, because both Apple and Kinko's knew how much innovative and useful courseware was being developed for Apple computers in academic communities throughout the country. And both felt that there should be a better way than word of mouth of publicizing these valuable resources and making them widely accessible.

Beside giving institutions and individuals a means of distributing their courseware, the Academic Courseware Exchange is also a resource bank of Apple II and Macintosh® programs that institutions and students can purchase at low cost.

### *Our Statement of Direction*

The Kinko's Academic Courseware Exchange enables the higher education community to purchase academic software for Apple personal computers at prices comparable to textbook prices. In doing so, it offers courseware developers the opportunity to market and distribute their products efficiently and economically.

The Academic Courseware Exchange seeks out high-quality academically oriented applications for Macintosh and Apple II computers, and makes them easily and inexpensively available to educators on a nationwide basis. Our goal is to encourage the use of computers in those aspects of education where the interactive nature of computer-based instruction makes it a more effective teaching instrument than other methods.

The Academic Courseware Exchange is a service company with two sets of valued customers: buyers and developers. We link those who have software to share with those who need it to teach their classes more effectively, providing both groups with the resources of Kinko's nationwide network of retail outlets, marketing, and distribution services.

For buyers, our goal is to provide:

- High-quality, discipline-specific academic computer programs, making them comparable to textbooks in cost and accessibility.
- A variety of software types (such as applications, templates for use with existing applications, and programming tools).
- Up-to-date information about Academic Courseware Exchange offerings.

For developers, our goal is to provide:

- An opportunity to be rewarded for their efforts with both financial return and peer acceptance.
- Promotion of their products to a wide audience.
- Fast, economical distribution of their packages.

A final goal is to be highly responsive to both groups—listening to them and responding to their needs.

## *1.2* Kinko's Copies, the Courseware Exchange Center

Kinko's Copies, headquartered in Southern California, is the Courseware Exchange center. Kinko's is particularly suited to manage the Academic Courseware Exchange because, with more than 400 Kinko's Copies locations in college towns nationwide, Kinko's main business is serving the duplicating needs of more than 600 colleges and universities.

Kinko's already offers the following services:

- "Professor Publishing," which lets faculty inexpensively and efficiently reproduce supplementary course materials and distribute them to their students.
- "Copyright Permissions Assistance Program," for professors and teachers who want clearance for use of copyrighted material in their courses.

- "Educational Reprint Service," which provides reprints and out-of-print articles from major academic journals such as the *Harvard Business Review*.
- Apple Desktop Publishing services.

Kinko's also publishes and distributes the *Academic Courseware Exchange Catalog* and provides a full range of services for academic developers who want to make their programs available to the greater academic community, through the Academic Courseware Exchange program.

## *1.3* The Purpose of This Handbook

This handbook introduces the courseware developer to Kinko's Academic Courseware Exchange program and explains how to participate. In particular, you'll learn the following:

- Ways to create academic courseware.
- The range of Academic Courseware Exchange services.
- How to document, test, and price courseware.
- How to apply to the Courseware Exchange.

Kinko's hopes you will join the many students and faculty and staff members from colleges and universities throughout the country who are successfully distributing their programs through the Academic Courseware Exchange.

# 2

# *CREATING COURSEWARE*

# CREATING COURSEWARE

## 2.1 Introduction

Unlike commercial products, which tend to target much broader audiences, the courseware distributed through the Courseware Exchange is generally sold to very specific audiences, usually at prices under $30.

Courseware is needed in all areas of instruction, but especially in the courses that have large enrollments, such as freshman English, history, chemistry, biology, calculus, engineering, and business.

Such instructional tools can help students to do the following:

- Perform calculations and graph results.
- Create and experiment with models and simulations.
- Explore new ideas.
- Apply learned concepts.
- Sharpen basic communication skills (reading, writing, preparing presentations).

In addition, courseware can help students who need remedial work or special tutoring.

## 2.2 Types of Academic Courseware

Academic courseware can be classified into four categories: templates, applications, tools, and authoring tools.

### *Templates*

Templates are add-ons or files linked to commercial productivity tools—MacPaint® artwork for use in biology laboratories, for example, or Microsoft Excel spreadsheets to be used in cost accounting courses.

Another type of template includes applications written in uncompiled BASIC or Pascal that require the purchase of a BASIC or Pascal interpreter.

### *Applications*

Applications are self-running, interactive programs designed to offer instruction on or provide opportunities for exploration within a particular discipline. An example would be a simulation of the theory of relativity for a physics course.

### *Tools*

Tools are self-running programs designed to help students analyze or process information. Examples include a statistics processor for a social science course or a text-analysis program for English composition.

### *Authoring Tools*

Authoring tools include authoring or programming languages and debugging tools—utilities that help to create new applications or templates.

## 2.3 How to Develop Courseware

You can follow one of several paths when you develop courseware:

- Use a commercial application as a base and create a template.
- Use an authoring language.
- Use a programming language to create compiled code so the application is self-starting.

### *Using a Commercial Application*

The easiest way to develop courseware is to create a template using a commercial application such as MacPaint or Microsoft Excel.

### *Using an Authoring Tool*

Authoring packages have revolutionized academic software development. Using authoring tools designed for nonprogrammers, you can create discipline-specific instructional software in a fraction of the time it used to take.

In addition to the inexpensive, yet powerful authoring packages listed in the *Academic Courseware Exchange Catalog,* Kinko's recommends the following authoring tools for the Macintosh:

### Course of Action and CourseBuilder

Course of Action and CourseBuilder are visual, icon-based authoring systems. They are designed especially for people who have no knowledge of programming—but who would like to use a computer to help express a concept, simulate a scientific process or phenomenon, or lead students through a tutorial. While each boasts unique strengths, both of these flexible tools allow the lesson creator to plan the course on the Macintosh screen and to run the course at any time just as the student would see it. They also have options available to control interactive videodisc from the lessons. In addition, both of these tools offer excellent demonstration disks.

Course of Action
Authorware, Inc.
(612) 941-5752

CourseBuilder
TeleRobotics International, Inc.
(615) 690-5600

### VideoWorks Interactive

VideoWorks II, successor to the original VideoWorks, has received reviews as the best two-dimensional animation program on the microcomputer market. A new version of the product, VideoWorks Interactive—designed specifically to include the capabilities for interaction, branching, and tracking so critical to creating quality courseware—is now available. In addition, MacroMind has also created an extension to HyperCard™ that allows developers to use VideoWorks II animations from within a HyperCard stack.

VideoWorks Interactive
MacroMind
(312) 871-0987

### HyperCard

HyperCard is a toolkit for manipulating, customizing, and accessing information. Using a "card" metaphor, it allows users to organize content by association and context. The result is a network of linked information that allows students to learn by following their interests.

In addition, HyperTalk™, the English-like scripting language of HyperCard, offers developers a great deal of flexibility, which lets them utilize various types of interaction with students. HyperCard has also been reviewed as an excellent controller for interactive videodisc and multimedia presentations.

HyperCard
Apple Computer, Inc.
(408) 996-1010

**Guide**

Guide is the first commercially available "hypertext" system produced for a microcomputer. It allows you to organize documents nonsequentially based on relationships between topics. After the documents and links are prepared by a professor, Guide provides an environment to help students navigate through the large bodies of information, exploring the information as they wish. Hot spots or "buttons" within a document link to related information in either the same document or other related documents.

Guide
Owl International
(206) 747-3202

***Using a Programming Language***

Finally, you can create courseware with one of the many programming languages, of which BASIC, Pascal, and C are the most popular. Generally, when courseware is developed in this manner, a "content expert" works with a programmer throughout the project. Many schools (such as Stanford, Dartmouth, and Drexel) have established faculty courseware development programs to coordinate these efforts.

If you want to develop courseware using programming languages, it is essential that you obtain a free, no-obligation developer's packet from:

Apple Developer Relations
20525 Mariani Avenue, M/S 27W
Cupertino, CA 95014
(408) 973-4897

The packet includes critical information on technical documentation, development languages, and utilities, as well as information on the use of Apple-licensed utilities such as the Finder™.

The packet also describes Apple's Certified Developer Program and includes an application form. Certified Developers receive special pricing on software development equipment. They also receive mailings of technical information, product announcements, and other information essential to the development community.

# 3

# *KINKO'S ACADEMIC COURSEWARE EXCHANGE*

# *KINKO'S ACADEMIC COURSEWARE EXCHANGE*

## *3.1* Introduction

The goal of the Academic Courseware Exchange program is to distribute software developed for the Apple II and Macintosh computers by higher education institutions and their faculty, staff, students, and other affiliates of higher education.

## *3.2* Academic Courseware Exchange Services

Through the Courseware Exchange, Kinko's provides the following:

- Software and documentation production and packaging.
- Nationwide marketing.
- Order processing and distribution.
- Site licensing of software to colleges and universities.
- Payment of royalties to developers.
- Limited warranty and replacement of damaged media and documentation for users.
- A conduit for feedback from users to developers.

### *Software and Documentation Production and Packaging*

Kinko's takes care of all production of your courseware package. The software is copied onto high-quality disks and labeled with Courseware Exchange labels that clearly show the program's title and author. The documentation is copied two-sided on 8.5- by 11-inch paper with the highest-quality Kodak and Xerox copiers. It is then bound with a disk holder in a durable plastic cover that identifies your courseware as part of the Academic Courseware Exchange program and protects the package during shipping—and later, in students' backpacks.

### *Nationwide Marketing*

At the time of publication, the marketing plan for the Courseware Exchange included the following strategies. Call the Courseware Exchange office for current details on marketing.

- Catalog distribution takes place twice a year: in the spring and fall. Catalogs are distributed through direct mail to a qualified list of customers, at selected local and national educational computer conferences, and through Kinko's Copies stores.
- Direct mail is used to reach the Courseware Exchange's primary customers: university professors. The direct-mail pieces are designed to build awareness of the program, and have received awards for both design and response rate.
- National advertising in major academic journals such as the *Chronicle of Higher Education* and *Academic Computing*.
- Kinko's Copies stores merchandise the Courseware Exchange through catalog displays and posters. In selected locations, campus sales representatives are also available to demonstrate your courseware to interested parties.
- Toll-free customer service lines are available in the United States for courseware information, ordering information, or software support.
- National conventions are attended by Courseware Exchange personnel who demonstrate available courseware and work to build awareness of the Courseware Exchange program.

### *Order Processing and Distribution*

To keep costs down, Kinko's maintains a very low inventory of disks and documentation at the headquarters of the Courseware Exchange. But even with this low inventory, all orders are fulfilled within 12 days of their receipt—most orders within 5 to 10 days.

Users can request courseware in several ways: directly at Kinko's locations, through a toll-free telephone number, or by mail.

- **Single-copy orders** (orders of less than five of one package) are filled completely by the Courseware Exchange headquarters. Orders can be prepaid by check, cash, or credit card, or charged through a university purchase order.
- **Multiple-copy orders**, placed by faculty for class use, are sent to the local Kinko's store, for purchase by the individual students. Kinko's campus reps or Courseware Exchange office personnel assist faculty in specifying the proper titles and quantities for their classes.

If there is no local Kinko's store, the Academic Courseware Exchange will ship multiple-copy orders directly to a school's bookstore or campus computer center. Kinko's cannot, however, be responsible for prices on products sold this way, because of possible markups by third parties.

### *Site Licensing to Colleges and Universities*

In addition to single-user software licenses, Kinko's has the ability to sell a courseware package through a site license. While participation in the site-licensing plan is encouraged, it is entirely optional. Kinko's feels that such participation will enhance both the use of the software and the financial returns to the developer. Attributes of the site-license policy include the following:

- The license is for an unlimited number of copies of the disk(s) from one courseware package—the purchasing university is responsible for the distribution of the disk. The software can be distributed on disk or through electronic methods. Disk labels will be available from Kinko's for a predetermined price—the university does not have to use these labels.
- The site license is for an unlimited term.
- The distribution of the disk is limited to the affiliates (faculty, staff, and students) of one physical campus of a university or college.
- Documentation will be made available for sale at the documentation price (to be published in the catalog) through the nearest Kinko's Copies store. The documentation can be bought in any quantity (no minimums) by any affiliate of the university. It can also be resold by the university to affiliates of the university. The university does not get the right to reproduce the documentation.
- Updates that become available within 90 days of the purchase of the license will be included in the purchase price.
- A single site-license price is set for each package by the developer and Kinko's, and that price is published in the *Academic Courseware Exchange Catalog*.

### *Royalty Payments to Developers*

Kinko's will maintain accounts and track orders. It also will make royalty payments twice annually: by September 30 for January–June sales, and by March 30 for July–December sales. Complete information about determining royalties is contained in Section 5.2.

### *Limited Warranty Replacement of Damaged Media*

Kinko's assures its customers of the highest-quality products. In the event that a customer receives a disk with defective or damaged media or an incomplete manual, Kinko's will provide replacements at no charge.

### *User Support and Feedback*

In order to keep courseware as affordable as possible, it is sold through the Exchange "as is"—essentially unsupported. However, in its effort to offer customers the highest-quality products and integrity in service, Kinko's will serve as a liaison between developers and customers, who can call the Courseware Exchange's toll-free number to report problems. Kinko's will work with one individual from the developing institution who can respond to reports of bugs in the program or ambiguities in documentation. Users will not be referred back to the developing institution. Kinko's will work with users to solve problems.

### *Copy Protection*

To minimize production costs, software distributed through the exchange must not be copy protected. Prices are kept low to discourage copying, and a copyright notice that advises users that it is illegal to copy the software is included with each product.

# 4

# PREPARING YOUR PROGRAM

# PREPARING YOUR PROGRAM

## 4.1 Introduction

Kinko's Academic Courseware Exchange can only accept completed programs that have undergone thorough testing and that include documentation. To assist developers, we offer guidelines for testing and documenting a courseware package.

## 4.2 Recommended Courseware Testing

Courseware you want to distribute through the Courseware Exchange must be thoroughly tested. **It also must be field-tested for at least one term (quarter or semester) in a classroom situation.**

Developers whose courseware is accepted by the Courseware Exchange must agree to work to solve problems in good faith by taking the following actions:

- Responding to bug reports within at least two weeks.
- Fixing a "fatal error" within 90 days of first report (a fatal error may be fixable simply by a workaround or by a warning in the documentation).

Field-testing, performed by students and teachers who are using your product in a course, is one way to find and correct any bugs or errors.

The other is to perform structured tests under simulated conditions. Structured tests systematically evaluate each feature and function of the program and its documentation.

When submitting courseware to the Courseware Exchange, you will be asked to include a Courseware Test Summary Report with your application. The summary should include descriptions of both field-testing and structured testing performed on your product.

If you want to learn more about software testing, a good guide is *The Art of Software Testing*, by Glenford Myers, published by Wiley-Interscience, a division of John Wiley & Sons.

### *Field-Testing*

Field-testing must be carried out for at least one term (quarter or semester) by students who are using the product in a course.

### *Structured Testing*

There are three basic types of structured testing you will want to perform:

- Feature/Documentation comparison checks for accuracy in the documentation and completeness of the program feature descriptions.
- Function testing evaluates each function alone and in combination with others to see that it works properly.
- Destructive testing involves entering values beyond the boundaries given in the manual to see if and when the courseware "blows up," and creating error conditions to verify that adequate error-handling messages are provided and to see how well the system recovers.

Destructive testing usually accounts for most of the testing, as these percentages indicate:

| | |
|---|---|
| Feature/Documentation testing | 15 percent of test |
| Function testing | 25 percent of test |
| Destructive testing | 60 percent of test |

A fourth type of structured testing—regression testing—is usually performed after a bug has been found and fixed, to ensure that the bug fix has not created any new problems with features that have already been checked.

### *Setting Up Structured Testing*

To set up a structured test, first create a testing guide that you can follow systematically.

#### The Testing Guide

Using the documentation or the software itself, list every feature, function, command, and menu item in the program.

Then, next to each item on the list, create three boxes and label them F/DC, FNCT, and DSTR (for feature/documentation testing, function testing, and destructive testing). These boxes will help you keep track of the testing as you proceed.

If you derive your outline from the software itself, it should either be command or menu oriented, and should follow each path to its

last level. Interaction between menus should be documented; commands should be grouped by similar functions.

For example, your outline might look like this:

| F/DC | FNCT | DSTR | |
|---|---|---|---|
| [ ] | [ ] | [ ] | 3. Review Incoming Mail Menu |
| [ ] | [ ] | [ ] | Display Mail |
| [ ] | [ ] | [ ] | Skip to Next Entry |
| [ ] | [ ] | [ ] | Display Text |
| [ ] | [ ] | [ ] | Rename to Sender's Filename |
| [ ] | [ ] | [ ] | Rename to New Filename |
| [ ] | [ ] | [ ] | Delete |
| [ ] | [ ] | [ ] | Print Text |
| [ ] | [ ] | [ ] | Print Mail |
| [ ] | [ ] | [ ] | Print Log of Received Mail |
| [ ] | [ ] | [ ] | Print Log of Incoming Mail |

### Hardware Configuration

Use the hardware configurations that you will specify be used with the product. (Use as many different configurations as possible.)

### Test Conditions

Try to simulate, as closely as possible, the actual conditions under which you expect the product to be used—the type of user you expect, the amount of data you expect to be entered, the size of records, and so on.

## *Feature/Documentation Testing*

To make sure that the documentation describes the functions completely and accurately, ask these questions:

- Is the documentation accurate?
- Do all the menu items do what the manual says they do?
- Do the examples given in the manual work?
- Are there any omissions?
- Does the manual adequately address all foreseeable user questions?

- Is the approach suitable for the intended audience?
- Is the user expected to be computer literate?
- Are special skills or knowledge required that are not provided?
- Does the documentation clearly and simply communicate what a user must do?
- Are there misspellings?
- Is the manual consistent?
- Can you find the topics you want in the glossary and index?
- Can you find chapters and sections using the table of contents?
- Does any part of the manual assume that the reader knows something that is only explained later?

### *Function Testing*

When testing each function, do the following:

- Verify that data is not lost or destroyed during normal use of the system.
- Create scripts or test cases representing normal use by the intended audience.
- Test the limits of each function as they are defined in the manual.
- Test system security (passwords).
- Using the command summary from the manual, check combinations of functions.
- Measure the time it takes the computer to perform functions, to determine acceptable limits.
- Check to see that different features can interface with one another under all acceptable hardware configurations.
- Test all disk possibilities, including full disks, write-protected disks, empty disks, and unformatted disks.
- Test the product's interface with peripherals.
- Use upper- and lower-case letters and special characters.

### *Destructive Testing*

During destructive testing, you want to deliberately create error conditions by entering values beyond the boundaries given in the

manual. You can be as creative as you like; the idea is to really challenge the product. Try any or all of the following:

- Check to see whether error messages are given promptly, and how easy it is to recover from errors.
- Use the wrong disks (full, write-protected, empty, unformatted).
- Remove the disks from the drives.
- Test values that are "just out of range" and "way out of range."
- Neglect to hook up or turn on output devices.
- Select numbers and letters not listed in the menus.
- Enter bad data in entry fields (null entries, entries too long, mixed alphabetic and numeric characters, control characters, invalid pathnames).

### *Regression Testing*

You should perform regression testing whenever you have found and fixed a bug, to determine whether the bug fix has caused problems with other aspects of the program. (Changes and error corrections tend to be much more error-prone than the original program code.)

To perform a regression test, rerun at least one subset of previous tests.

### *Writing the Courseware Test Summary*

When testing is complete, write a brief summary of the testing program and submit it with your courseware. It should address both field-testing and structured testing.

## *4.3* How to Prepare Documentation

Good documentation is important to the usefulness of your product. It doesn't have to be long or arduous—some documentation may be limited to the mandatory disclaimer and copyright notice and a few short notes—but it should be sufficient to help users accomplish what is intended.

Apple computers are designed to make life easier, and your courseware should make teaching and learning a more pleasant experience. It helps if your manual is friendly and appealing, while still being complete and easy to use.

Use the guidelines that follow when you develop documentation for your courseware. They'll assist you in maintaining consistency and clarity and, if you have never developed software documentation before, will introduce you to a successful documentation method.

Also consider Kinko's as a resource; Kinko's will be pleased to help you resolve any issues regarding courseware documentation.

When you submit your courseware to Kinko's for distribution through the Courseware Exchange, Kinko's will also review your accompanying manual for general readability and clarity.

### *The Apple Model*

Your documentation should follow the example set by Apple Computer. As one of the first companies to recognize the need for a new approach to writing computer manuals, Apple is known for manuals that teach, entertain, and help people who are unfamiliar with or afraid of computers.

#### Who should write documentation?

Documentation is generally best written by someone other than the person who developed the software. The developer is often too technically knowledgeable or too immersed in the product to understand how to explain it clearly to others.

Documentation should be written, or at least be reviewed, by **writers**—people who understand how to communicate in written language. This increases the likelihood that the resulting documentation will meet the needs of users by being clear, sensitive, and consistent.

#### Why is consistency important?

Apple's writers have chosen to use a certain set of conventions so that all their documentation is consistent (see Appendixes). Because of this, a user can expect to find certain sections in all Apple manuals. A particular action is always called the same thing from one section to another and from one manual to another. This means that the user is not frustrated or confused by nebulous or inconsistent terms.

### *Submitting Documentation*

Documentation should be submitted as finished, pasted-up art ready for reproduction.

It is recommended that you produce your documentation on the Macintosh computer because of its integrated text and graphics and variable font capabilities.

Kinko's accepts finished, pasted-up, typeset-quality text (produced by an Apple LaserWriter® printer or equivalent) and graphics in 8.5- by 11-inch format. **Text should be set in the Times® font, using the type conventions shown in Appendix A.**

### *References*

The best models for writing documentation are the user manuals written by Apple for its products. Check out any of these:

- The MacWrite® user manual.
- The MacDraw® user manual.
- The AppleWorks® user manual.
- The AppleWriter™ user manual.

Also, for Macintosh developers, the user-interface guidelines in Apple's *Inside Macintosh*, published by Addison-Wesley, are useful.

For references for style, punctuation, and word usage, try the following:

- *Words into Type*, third edition (New Jersey, Prentice-Hall).
- *The Elements of Style*, Strunk & White, third edition (New York, Macmillan Publishing Co.).

Another reference that is helpful for writers of computer documentation is *How to Write a Computer Manual: A Handbook of Software Documentation,* Jonathan Price (Massachusetts, Addison-Wesley).

### *Writing Tips*

Einstein said, "Make everything as simple as possible—but no simpler." His words apply to writing computer manuals, because you want them to do the following:

- **Get your reader doing something quickly**. Readers are intelligent and impatient. Don't launch into detailed explanations of how your program works; tell them how to use it in a meaningful way.
- **Be graphically appealing.** The way you position information on the page has a great impact on the readability of your manual and its usefulness as a reference tool.
- **Organize information that parallels the reader's thinking**, not computer logic.
- **Use clear examples** that users can relate to, and include tutorials.
- **Tell the whole story.** Your manual should be structured so that a reader can find the references needed to solve problems. Make sure that it includes a complete table of contents and the necessary "cookbook" material and appendixes. Also include references to user manuals for the computer or to other useful software manuals.

### *Target Audience*

Your intended audience will determine the form, content, and style of your manual. Once you know the nature of your audience, you can better anticipate what they need to get from your manual.

Generally, Apple courseware users fall into two groups: academic professionals and students.

#### Academic Professionals

These are administrators and faculty members in institutions of higher learning. They are concerned about the quality of teaching, their professional and institutional images, their budgets and sources of funding, and ways to inexpensively increase productivity on campus. Schools want to maintain an innovative and professional image.

#### College Students

College students are amenable to change and quick to try out new ideas. They are very curious and willing to take on new challenges. Typically, college students see the computer as a fancy gadget that will help them to do better in school and to prepare for getting a job after graduation.

Both groups are composed of intelligent, well-educated, busy people who probably don't know too much about how or why a computer works. They just want to use a computer to help them be more productive.

### *Manual Organization*

Most manuals will have the following structure:

- **Front**
  Disclaimer and Copyright Notice
  Title Page
  Table of Contents
  Introduction
- **Body**
  Tutorials
  Cookbook
  Advanced Tutorial or Examples (for complex applications only)
  Reference
- **Back**
  Appendixes
  Glossary
  Index

Your manual doesn't need to cover material included in the user manual that comes packed with the computer and its system software, such as information on the following:

- Unpacking and setting up the system.
- Computer parts and functions.
- Using the system.
- Care and handling.

### *The Front of the Manual*

#### Disclaimer and Copyright Notices

The need to include a disclaimer and copyright notice is a legal fact of life; all product manuals must contain this "fine print." The disclaimer and copyright notice you provide protects your manual and software from copyright infringement and helps protect you from liability. You should look at various software manuals to determine

what sort of language you might use. Consult the Participation Agreement for more information regarding the copyright and disclaimer notice.

### Title Page

Include on your title page:

- Name of software.
- Name of developer(s).
- Affiliated institution or school name.
- Software version or release date.
- Acknowledgments.

### Table of Contents

The table of contents is called "Contents" in Apple manuals. Yours should include all headings and subheadings appearing in text, and their corresponding page numbers. See an Apple manual for examples of how to structure the table of contents and how to capitalize the titles of the entries.

### Preface

Rather than calling the first text section the "Preface," you might prefer more friendly, descriptive titles such as:

- "About This Manual"
- "About [THIS PROGRAM]"
- "How to Use This Manual"
- "Using [THIS PROGRAM] to Teach [SUBJECT]"
- "Teaching [SUBJECT] with This Courseware"

Write a sentence or two that describes your courseware, suggestions on how to use the manual, what kind of background teachers and students should have to get the most out of your courseware, and how the manual is structured.

You should also include information about how your courseware was used at your school or other schools, your experiences using your courseware in teaching, student's reactions to your courseware, or how else it may be used in teaching.

The preface should be friendly, and above all else, SHORT.

### Introduction

An introduction includes a description of your courseware, what it can do for the reader, how it works, and what equipment the reader needs.

The introduction should have a friendly title, such as:

- "Getting to Know [YOUR PROGRAM]"
- "Introducing [YOUR PROGRAM]"

### *The Body of Your Manual*

The body of a manual may consist of several parts :

**Tutorial**, called "Getting Started" or "Learning [Your Program]": material that guides readers step-by-step through the basics of your courseware.

**Cookbook,** called "Using [Your Program]": a collection of task-oriented, "How to" pages.

**Exercises:** a sampling of uses of your program's basic features, and perhaps a tutorial introducing users to more advanced features (if your courseware includes complex functions).

**Reference Material,** including menu summaries, shortcuts, and special considerations: a concise summary of commands and functions.

### The Tutorial

The manual should include a basic tutorial to show the user how to do something useful immediately. The tutorial doesn't need to contain all of the features of your courseware, just the basics that will get a new user going quickly. Your tutorial should:

- Guide the reader step-by-step.
- Point out what the screen shows at each point in the process.
- Anticipate confusion.
- Remind gently.

- Explain the significance of each step.
- Get the user out of trouble—or, preferably, prevent the user from getting into trouble.

Writing a tutorial usually requires a lot of rewriting. As you go, you have to keep asking yourself the question: "But what do they need to know first?" You'll find that you think of things you should have mentioned earlier—and things you should have mentioned before *those* things....

If appropriate, offer some sample data files, so users don't have to create anything before they can use the courseware.

Tutorials should consist of several sections, each building on the skills developed in the previous one. Each section should take no more than about 15 minutes to work through.

See the tutorial sections in Apple manuals for ideas on how to structure a tutorial.

### The Cookbook

This portion of the manual is dedicated to instruction on how to do specific tasks or processes. It is more than a command summary, because it actually shows the steps needed to achieve a certain result.

The cookbook section should constitute the largest portion of your manual, approximately 50 percent. This is because you can include material here that would ordinarily end up in a tutorial or be buried in a reference section.

### Reference Material

The reference section provides readers with information about commands or basic elements of your courseware. Generally, readers will use the cookbook section for a specific task and the reference section for information about the structure and operation of the courseware. However, don't include highly technical or theoretical information here.

A summary of menu commands is a very helpful inclusion in the reference section. Look for examples in the Apple manuals.

### *The Back of the Manual*

The back of your manual should consist of an index, a glossary, and any appendixes you might want to include.

#### Index

Includes topics and ideas, and shows the page numbers on which they are covered.

#### Glossary

Includes definitions of words and terms that may be unfamiliar to some readers; acronyms, commands, and phrases that apply only to your courseware. The best way to find out what people find puzzling is through testing—try out your manual and courseware on some novices and ask them what is unclear.

#### Appendixes

An appendix is a collection of supplementary material that may be appropriate only for some small portion of your audience and not necessary for using your courseware. If you have some material that doesn't seem to fit with the organizational model for your manual, it probably belongs in an appendix.

Some possible topics:

- File formats for programmers.
- How to transfer output to other tools.
- Courseware specifications
- Troubleshooting.
- Converting files from one format to another.

## *4.4* Legal Considerations

### *Ownership*

The developer maintains ownership of the courseware package. Kinko's Courseware Exchange acquires a right to distribute and market the package, nonexclusively, throughout the world.

### *Copyright*

It is very important that you include a copyright notice on software when your product is first distributed. If you are using Apple system software (such as the Finder or ProDOS®), it is important that you also include Apple's copyright notice.

You should include such copyright notices on your disks, as well as in any manuals for your courseware.

The Academic Courseware Exchange Participation Agreement asks that the developer provide copyright registration information for all material to be distributed. Using Form TX, you register copyrights through the U.S. Copyright Office, Library of Congress, Washington, D.C. 20559. An attorney can address any further questions you may have on this subject.

### *Trademarks and Service Marks*

To protect a trademark by law, a company registers its name, logo, and product names. The registered trademark symbol is a circled, superscript small capital R: ®. The ™ mark indicates an unregistered trademark. It represents a company's common-law claim to a right of trademark, or a trademark for which registration is pending.

The following sections provide trademark and service mark use guidelines for references to Apple products. Be sure that your software and documentation complies with these guidelines.

#### Apple Logo and Apple Name

The Apple logo (the colored Apple and solid-color Apple silhouettes) is a registered trademark and may not be used by anyone except Apple. The logo is the property of Apple Computer, Inc.

The word *Apple* (when it refers to a product and not the company) is a registered trademark. The first time Apple appears in reference to a specific Apple product, it should be immediately followed by the registered trademark symbol. For example:

*Apple Computer, Inc. is proud to announce the Apple® IIc.*

The registration symbol correctly follows the use of *Apple* when it refers to the product, not the company. It follows the word Apple, not IIc, because the IIc is a model designation, not part of the trademark.

### Other Apple Trademarks

If the name of an Apple product is a registered trademark, use the ® symbol with the product name the first time it appears. Check any product you refer to for a trademark and use the vendor's credit line when referring to it.

# 5

# *HOW TO SUBMIT COURSEWARE*

# HOW TO SUBMIT COURSEWARE

## 5.1 Submitting Courseware

When you have a software package that is tested and well documented, you are then ready to prepare your materials to submit to Kinko's. This section of the *Developer's Handbook* covers the final stages of preparing your software submission:

- Software Pricing
- The Participation Agreement
- Kinko's courseware evaluation process

## 5.2 Pricing Your Product

One of your most important decisions will be how to price your courseware, because the price you charge has an effect on the way your software will be used and the royalties you will receive. Kinko's has the ability to sell your courseware two different ways—through a single-user license and through a site license. Determining a single-user price is relatively easy and methodical; the site-license pricing is more challenging and less defined (it is also optional).

In all cases, pricing must include all of Kinko's administrative, marketing, and production costs, the royalties you will receive, and a fair profit margin for Kinko's.

### *Single-User Pricing*

The courseware sold through the Courseware Exchange is priced so that professors will be encouraged to recommend that students purchase it. Generally, this recommendation is most likely to be made for courseware that is priced under $30. In special instances, Kinko's will allow prices to go as high as $40.

### Suggested Pricing

Pricing is primarily your decision. As a general rule, Kinko's suggests the following guidelines:

- Template-based programs sell for approximately $9 to $18.
- Applications sell for approximately $12 to $25.
- Tools sell for approximately $18 to $28.
- Authoring tools sell for approximately $25 to $35.

When determining your pricing, consider how many hours the program will be useful to a student—does the program cover a specific topic only or will it be useful throughout a course?

If you do not wish to take a royalty, Kinko's will sell the courseware as inexpensively as possible.

These are guidelines only; however, your pricing recommendations may affect Kinko's ability to cover costs—and, consequently, to carry your courseware.

### Determining a Price

The price of an Academic Courseware Exchange package is made up of three things: production costs, Kinko's margin, and the developer's royalty. In order to cover all advertising and administrative costs, and to make a fair profit, Kinko's must maintain at least a 30 percent margin on all programs. In order to maintain low prices, Kinko's also requires that no developer royalty exceed 40 percent of the retail selling price. The selling price must not exceed $40, and special approval is required for prices over $30.

### Production Costs

| | |
|---|---|
| a. Number of disks _____ X $1.50 = | _____ |
| b. Number of pages (sides) of documentation _____ X $0.05 = | _____ |
| c. Binding | $1.50 |
| d. Packaging, disk duplicating, and handling | $4.00 |
| e. Total production costs (add a, b, c, and d) | _____ |

### Calculating a Price

Determining a price is an iterative process, which goes as follows:

1. Determine an amount you would like to receive per package sold.
2. Plug it into this formula to determine the selling price: (Production costs + Developer royalty ) / (.70)
3. Ask yourself the following questions:
   - Is the royalty under 40 percent of the selling price?
   - Is the selling price under $30 (or $40 in special cases)?
   - Is this a fair price for the end users of the software?

4. If you answer yes to these three questions, then you have a price. If you answer no to one or more of the questions, then reevaluate your approach to Step 1 and work through the process again.

**Developer's Recommended Single-User Price**

Developer Royalty ($) _____

Developer Royalty (%) _____

Selling Price ($) _____

Remember, the price does not become final until it is approved by Kinko's.

### *Site-License Pricing*

Kinko's site-license policy gives colleges and universities the chance to easily purchase the right to make unlimited copies of a courseware package for use by their affiliates (students, professors, and administrators). The policy gives developers increased distribution of their programs at other campuses and a flexible and profitable pricing structure.

**Determining a Site-License Price**

The process of determining a site-license price is less methodical than that of deciding on a single-user price. The site-license price is primarily determined by the developer with assistance from the Academic Courseware Exchange office. Things to consider when determining a site-license price include:

- Participation is at the discretion of the developer.
- The price for the license can range from $150 to $5,000 and is based on the expected usage for the program at an "average" university.
- The initial sale price is split 50/50 between Kinko's and the developer. The developer also receives 2¢ per page of documentation sold to users of the software under the site license. In cases where the developer does not want a royalty, Kinko's will sell the site license for as low as $150; site licenses with royalties start at $300.
- Updates are included for 90 days; therefore, it is possible to resell a license to a university when a major upgrade becomes available.

- The expected use of the program in a class—a tool with a few hours of instructional value vs. a tool useful to all students throughout their college education.
- The number of people the program is expected to be useful to—a graduate-level package vs. one designed for a required undergraduate class, for example.
- The average size of the university you expect the program will be used at.
- The number of copies of the program that would be purchased in lieu of purchasing a site-license—would a professor require every student to purchase the program or would he or she be more likely to put two copies "on reserve" at the microcomputer center.
- A university will base its decision on two things: the amount a site license will cost vs. the cost of purchasing individual packages and the legal liabilities it can avoid by purchasing a site license.
- Prices between $300 and $1,200 are much more "palatable" to purchasers than prices over $1,200.
- The $5,000 maximum is there for an exceptional package and is probably higher than any university is willing to pay.

**Developer's Recommended Site-License Price**

| | |
|---|---|
| Selling Price ($) | _____ |
| Developer Royalty (%) | 50% |
| Developer Royalty ($) | _____ |

**Site-License Documentation Price**

*Production Costs:*

| | |
|---|---|
| a. Number of pages (sides) of documentation _____ X $0.05 = | _____ |
| b. Binding | $1.50 |
| c. Courseware Exchange Materials | $1.50 |

*Developer Documentation Royalty:*

d. Number of pages (sides)
of documentation _____ X $0.02 = _____

e. Total documentation price
(add a, b, c, and d) _____

## 5.3 The Participation Agreement

Each piece of courseware submitted for distribution through the Academic Courseware Exchange must be accompanied by a completed, signed Participation Agreement. This agreement does the following:

- Helps Kinko's to catalog and sell the courseware.
- Clarifies the business arrangement with Kinko's.
- Sets preliminary pricing for the courseware.
- Sets the terms and conditions designed to protect the developer and Kinko's.

### *Section I: Courseware Information*

This section includes the following:

- Title of the courseware.
- A list of the name(s) of the individuals and/or organizations that developed the courseware, as they should appear in advertising.
- Version, date, copyright date, and copyright holder.
- A copy of the copyright registration application (copyright may be pending) or for public-domain software, a signed statement from the developer/institution relinquishing any claim on the property.
- A general and specific academic subject area.
- Courseware classification (template, application, tool, authoring tool).

### *Section II: System Requirements*

When listing system requirements, mention everything that's needed to run the courseware, including:

- System (Apple II or Macintosh).
- All hardware and software configurations under which the program was tested (and whether each works).
- Compatibility with printers and other related peripherals.
- Number of disk drives (including a hard disk, if necessary).
- Minimum and maximum memory.
- Operating system or Finder version (for example, ProDOS or Macintosh Finder version 4.1).
- Languages or applications necessary to run the program (for example, MacDraw, AppleWorks, or Microsoft BASIC).

Be sure to identify other special requirements—expansion cards, mouse, joystick, or color monitor, for example.

### *Section III: Pricing Worksheet*

This section covers both site-license and single-user pricing for your package. This pricing is approximate and must be approved by the Courseware Product Manager and Kinko's before the package is available.

### *Section IV: Courseware Product Manager*

Provide the name, address, and title of someone who will act as **courseware product manager.** He or she must be an affiliate of an institution of higher learning (faculty, staff, or full-time consultant) and be able to take responsibility for business communications with Kinko's.

### *Section V: Technical Support Contact*

Identify a technical support contact to whom Kinko's can make requests for technical support or report problems and documentation errors. This person should agree to provide technical support as specified in the Participation Agreement Section XI.

### *Section VI: Royalty Payments*

Section VI asks for information on the payment of royalties to the developer or institution. Returns on sales of this courseware will be

paid to the ONE designee indicated on the form. It is not possible for Kinko's to split the payments of returns among several parties. Please include the name EXACTLY as it should appear on the check and information about where Kinko's should send the payment.

### *Section VII: Testing Summary*

This section asks for information on the use of the program and for the structured testing report. It also covers the proofreading of the documentation.

### *Section VIII: Software Description*

Besides providing a brief description for catalog information, please include a detailed description of roughly 200 words with as much of the following information as possible:

- Courseware target audience.
- Courseware objective.
- Type of instructional format (simulation, problem-solving, drill, or other).
- Number of lessons and/or amount of time required.
- Examples of usage.
- Locations where the courseware has already been used.
- Reviews of the courseware, including journal title, date, and page number. (Attach the review, if possible.)
- A list of keywords for possible inclusion in a computerized courseware database.

This information will help Kinko's represent the courseware accurately in the catalog.

### *Section IX: Materials Submitted*

This section summarizes the materials that need to be submitted.

- 2 copies of each disk, each labeled "MASTER."
- Master copies of documentation.
  You must submit pasted-up, finished artwork (stats are acceptable) ready to be produced. Artwork will not be returned.
- An electronic version of the documentation on a third disk.

### *Section X: Processing Fee*

With each piece of courseware you submit to the Courseware Exchange, include the nonrefundable processing fee:

- $25 for courseware with no developer return.
- $100 for all other courseware.

Kinko's will accept a check, money order, or institutional purchase order.

### *Section XI: Terms and Conditions*

Section XI identifies the terms and conditions of the contract with Kinko's. It should be carefully reviewed, signed, and dated by an authorized representative who is affiliated with the developer/institution.

### *The Completed Submission*

Send your complete submission, in one package, to:

Kinko's Academic Courseware Exchange
Attn: Developer Sales Manager
4141 State Street
Santa Barbara, CA 93110

## *5.4* Kinko's Evaluation Process

The process after Kinko's receives a submission can take from three weeks to eight weeks—depending on the backlog. In all cases, you should receive a receipt within two weeks confirming your package's arrival. The process Kinko's uses in processing your submission is well-defined:

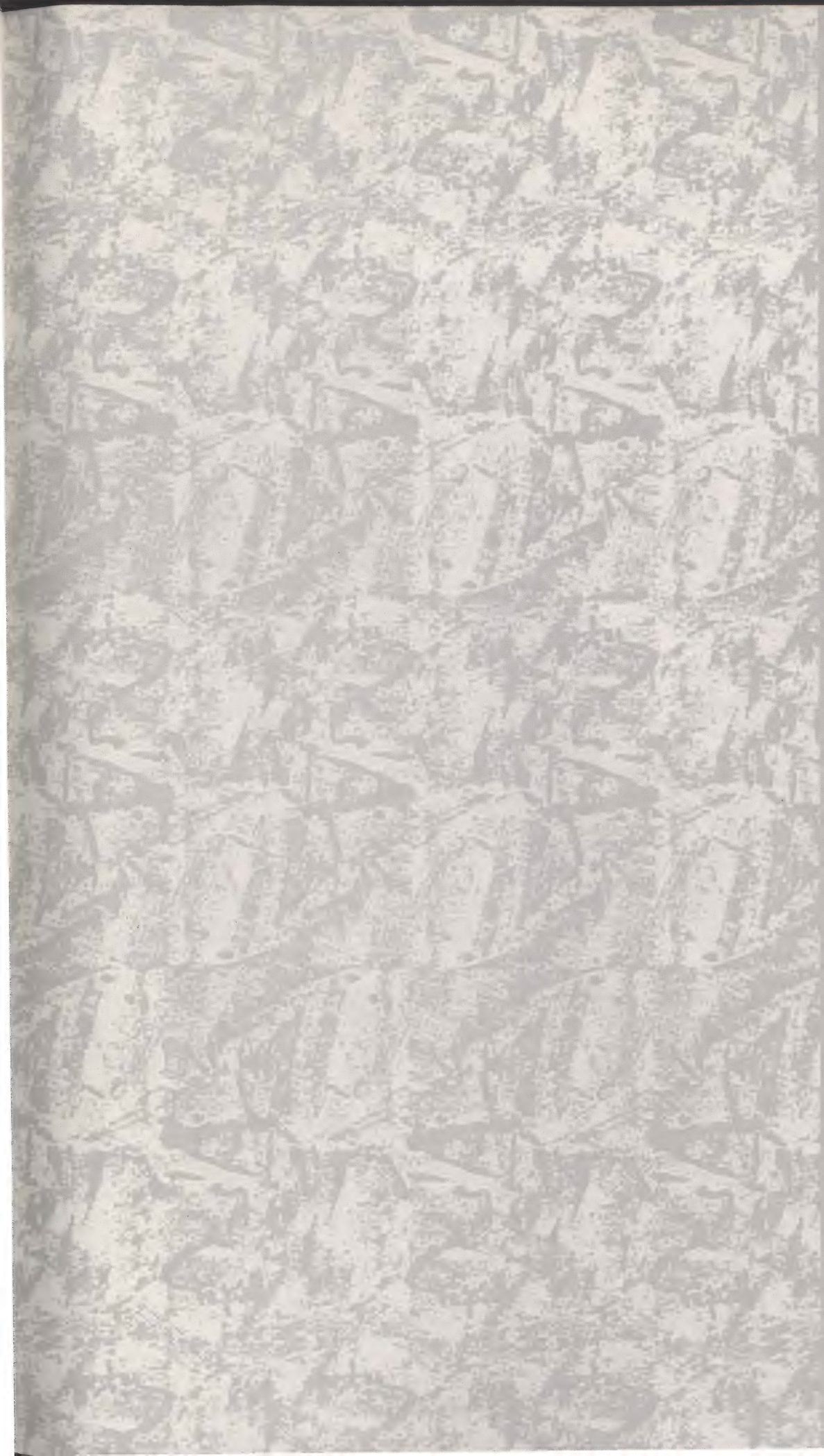
- The submission is checked for completeness.
- The subject area the software covers is checked to ensure that it is applicable to higher education.
- The software is evaluated against the documentation for functional integrity.
- The information from the Participation Agreement is entered into the Kinko's software database.
- The catalog entry is prepared concurrently with the sample package—disk and documentation.

- The pricing is verified and the sample package, Pricing Approval, and Catalog entry are sent to the Product Manager for final approval.
- Upon receipt of the signed Pricing Approval and catalog entry, the contract is signed by Kinko's and a copy is mailed to the Product Manager.
- At that point, the software package becomes available for sale.

If any questions arise during this process, Kinko's will contact the Product Manager. If there are any problems, Kinko's will do its best to assist the developer to rectify them.

## *5.5* Questions?

Questions about the Academic Courseware Exchange? The Participation Agreement? The terms and conditions? Call Kinko's **Developer Sales Manager** at 800-235-6919 (in California, 800-292-6640).



# 6

# APPENDIXES

# *APPENDIX A: STANDARD TYPE-STYLE CONVENTIONS*

If you are using a Macintosh to prepare your documentation, please use any of the following fonts (type specifications). The names New York/Times, Geneva/Helvetica®, and Chicago refer to fonts in the Macintosh Font menu. They are all available using any word processor for the Macintosh.

| For | Use | Example |
|---|---|---|
| All text | New York/Times<br>12 point | **Text** |
| Headings | Boldface<br>Capital letters<br>12 or 14 point | **ABOUT [YOUR PROGRAM]** |
| Subheadings | Boldface<br>Upper- and lower-case<br>12 point | **Preparing Your Information** |
| Headers/ Footers | Geneva/Helvetica<br>9 point | **Macintosh Pascal 3** |

# *APPENDIX B: STANDARD USAGE CONVENTIONS*

| For References to: | Use | Example |
|---|---|---|
| Program Names | Plain Text | Macintosh Pascal |
| Menu/Commands | Chicago<br>Use "Choose" instead of "Select" | Choose *Print* from the *File* menu |
| Words in Dialog Boxes and Buttons | Chicago | Click *OK* |
| Keyboard Keys | Plain Text<br>Capitalize key name | Press the Enter key |
| Section Names | Plain Text<br>Use quotation marks<br>Capitalize section name | ...in the "Reference" section |

# *APPENDIX C: STANDARD DOCUMENTATION TERMS*

**Alert box.** An *alert box* contains a *message* that does not require information from the user. Refer to an alert box simply as a *box.*

**Apple II, Apple II Plus, Apple IIe, Apple IIc, Apple IIGS, Apple IIs, Apple II Pluses, Apple IIe's, Apple IIc's, Apple IIGS's.** (Note plural forms.)

**Authorized Apple dealer.** Don't use *dealership* or *Apple-authorized dealer.*

**Boot.** Don't use. Use *start up* or *switch* instead.

**Built-in disk drive.** Use instead of *internal disk drive.*

**Bus.** Use *network.*

**Cable.** The cordlike thing that goes between two pieces of hardware. *Cords* are cordlike things that go to power sources (power cords).

**Check box.** Usually represents alternative options. If one thing is checked, another is unchecked.

**Choose.** Limit the use of *Choose* to menu commands.

**Click.** The action of briefly pressing and releasing the mouse button. Use *click in* for windows. Don't use *click on* for anything; use just *click* instead. For example: *Click Cancel.*

**Close.** You *close* windows and documents, not icons

**CLOSED-APPLE key.** Use *SOLID-APPLE key.*

**Command key.** On the Macintosh, the key with the clover-leaf symbol.

**Cursor.** For Apple II, use *cursor.* For Macintosh, use *insertion point.*

**Desk accessories.** Capitalize the names of individual desk accessories: *Clock, Clipboard, Scrapbook*

**Desktop.** One word (not desk).

**Dialog box.** Appears on the screen to request or give information. Just use *box* unless you feel the need to be more specific.

**Disk.** Not *diskette, floppy, floppy diskette, microdiskette, or anything else.*

**Document.** For Macintosh, a document is anything the user creates with a Macintosh application. For Apple II, *file* is OK.

**Double-click.** Hyphenated when used as a verb. For example: *Double-click the icon.*

**Drag.** The action of positioning the mouse on something, pressing and holding down the mouse button, moving the mouse, and then releasing the mouse button. Always use *drag* in reference to objects on the screen. For example:
*Drag the icon across the screen.*
Don't use *Drag the mouse.*

**e.g.** Use *for example* instead.

**80-column text card.** Not *80-column card* or *80-column board.*

**Enter.** Don't use *enter* when you really mean *press* or *type. Enter* is appropriate when referring to data.

**Enter key.** Note capitalization.

**Error message.** For Macintosh, use *message*. For Apple II, *error message* is OK.

**ESC, ESCAPE.** Use *ESC key* when referring to the key.

**Folder.** For Macintosh, a *folder* may contain documents, applications, and other folders, allowing you to organize information hierarchically. For Apple II, use *subdirectory.*

**Font.** Use instead of *typeface*.

**Format.** For Macintosh, use *initialize* instead.

**Input.** Never use as a verb. Use *enter* or *type*.

**Mac.** Don't use. Use *Macintosh.*

**Mice.** Don't use. Use *mouse devices*.

**Modem port.** Note capitalization. Don't use *Phone port.*

**Scroll box.** Two words.

**Shift-click.** Use a hyphen to denote a combined action (as in *Command-X*). Note capitalization and hyphenation.

**Space bar.** Lower case, two words.

**Start up.** Two words when used as a verb. For example:
*Start up* the Macintosh.

**Startup.** One word when used as an adjective. For example:
Insert the *startup* disk.